Advancing Science Through High Performance Computing

Esmond G. Ng
(egng@lbl.gov)

National Energy Research Scientific Computing Center
Lawrence Berkeley National Laboratory

The National Energy Research Scientific Computing Center

- The National Energy Research Scientific Computing (NERSC) Center is a national supercomputing facility of the Office of Science at the US Department of Energy (DOE).

- 25th anniversary in 1999.

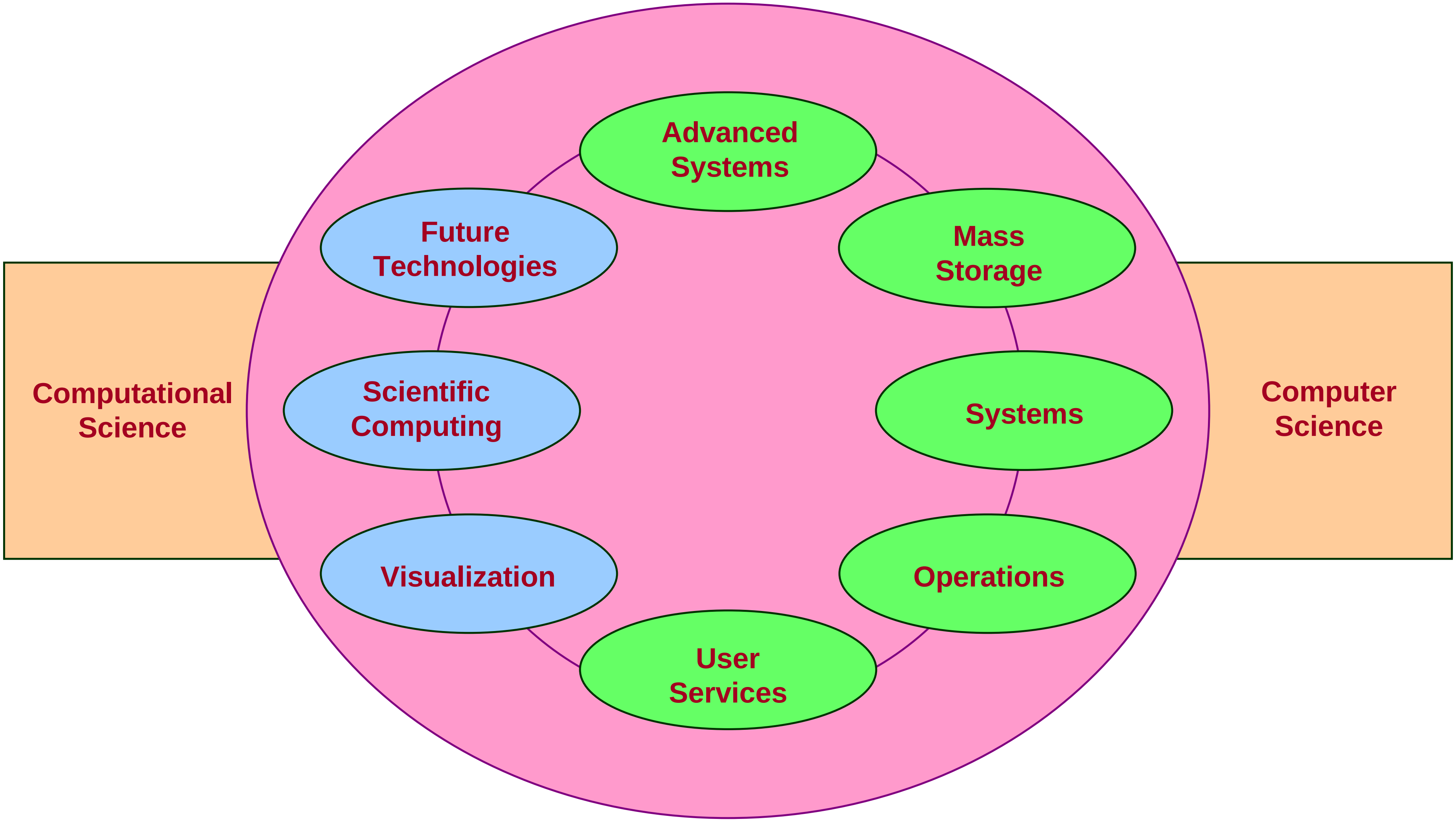
- NERSC provides unclassified, open computing resources; serving >2,000 users in basic science disciplines that are relevant to DOE mission.
- NERSC's mission is to advance science by making new scientific discoveries using high-performance computing.

Reengineering Large-Scale Scientific Computing

- In 1995-1996 DOE and NSF competitively re-examined the role of centers.
  - Rapidly changing technology.
  - Growth of computational approaches in all disciplines.
    - as important as theoretical and experimental research
  - Facilities alone are necessary but not sufficient.

- NERSC's New Model: Major Facility + Intellectual Services
  - The center serves as the working interface between computer science and physical science.
  - Creation of two departments:
    - High Performance Computing Department
    - High Performance Computing Research Department
  - The research department
    - develop new methods, algorithms, and tools via medium to long term collaborations with scientific user community
    - success stories
      - materials science (NERSC, ORNL, AMS, BNL)
      - cosmology (NERSC, LBNL, CalTech, UC Santa Barbara, UC Davis)

Intellectual Home of NERSC



NERSC Future Technologies Group

- Mission:
  - Research and development on next-generation infrastructure for scientific computing.

- Major focuses:
  - Clusters.
  - Tools for parallel programming (ACTS).
  - High performance access to remote storage and network-aware applications (DPSS, Netlogger).

Clusters

- Clusters of PC's:
  - Competitive today with traditional servers for small to medium sized problems.
  - May replace large supercomputers in 2-3 years.

- Advantages:
  - Low hardware cost.
  - Seamless desktop to teraflop integration.
  - Flexibility in configuration --- can be tailored to users.
  - Development platform for large systems.
  - Parameter studies for subproblems.

- Disadvantages:
  - Very expensive to setup and maintain --- expertise required.
  - Environment not very robust.

- SLAC-NERSC collaboration:
  - Helped SLAC with purchase and configuration of PC cluster.

LBNL Cluster Activities

- BLD --- Berkeley Lab Distribution.
  - Plug-and-play software distribution for scientific clusters (Release 1 in 1H2000).

- High performance *standardized* communication for clusters.
  - M-VIA: Virtual Interface Architecture (VIA) for Linux (Release 2 in 12/99).
  - MVICH: MPI for VIA (Alpha release).

- Scalable system software for large production Linux clusters. Nascent multi-lab/multi-agency effort.
  - Addresses possible lack of vendor support for very large systems in 2-3 years.
  - Berkeley, Argonne, Los Alamos have formed a close collaboration.
  - Tutorial on production Linux clusters at SC99.

- Both BLD and M-VIA will benefit SLAC applications.

NERSC Scientific Computing Group

- Missions:
  - Interact and collaborate with the scientific community in research and development on computational areas that benefit DOE and the nation.
    - design and implementation of highly efficient computational kernel algorithms for current and future NERSC applications
    - develop state-of-the-art methodologies and strategies for computational sciences

- Major focuses:
  - Numerical linear algebra.
  - Adaptive refinements for unstructured meshes.
  - Materials Science.
  - Astrophysics.
  - Earth/environmental sciences.

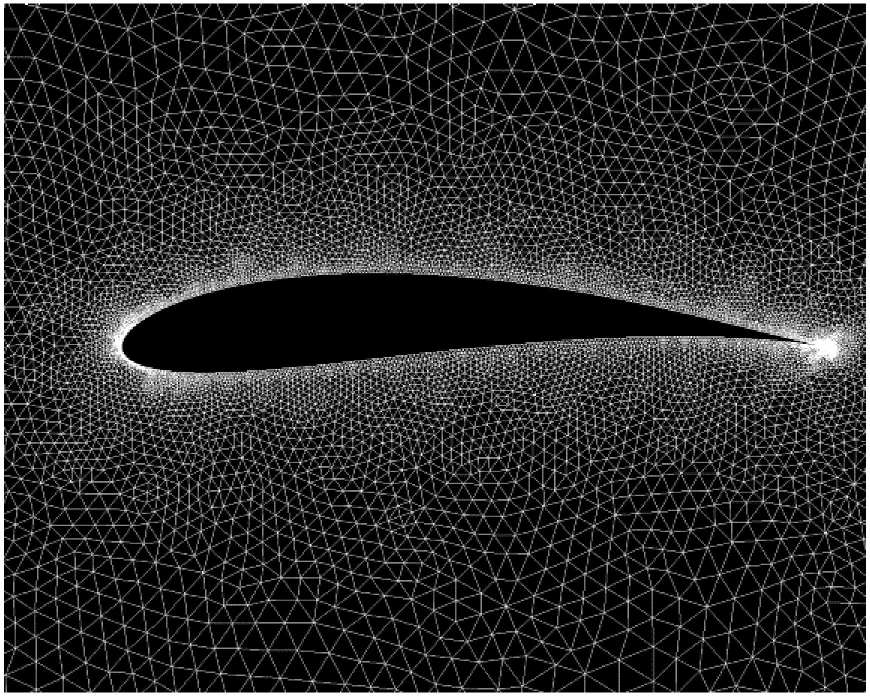
Scientific Computing Collaborators

- Proximity to researchers at nearby top universities.
- Stanford University, Scientific Computing and Computational Mathematics Program (SCCM).
  - Gene Golub, Fletcher Jones Professor of Computer Science
    - eigenvalue and singular value computations
    - iterative methods for solving systems of linear equations
- University of California, Berkeley, Computer Science Department.
  - James Demmel (adjunct appointment @ LBNL/NERSC)
    - numerical linear algebra algorithms (LAPACK, ScaLAPACK)
  - Jonathan Shewchuk
    - computational geometry, mesh generations
- University of California, Davis, Computer Science Department.
  - Zhaojun Bai
    - eigenvalue computations

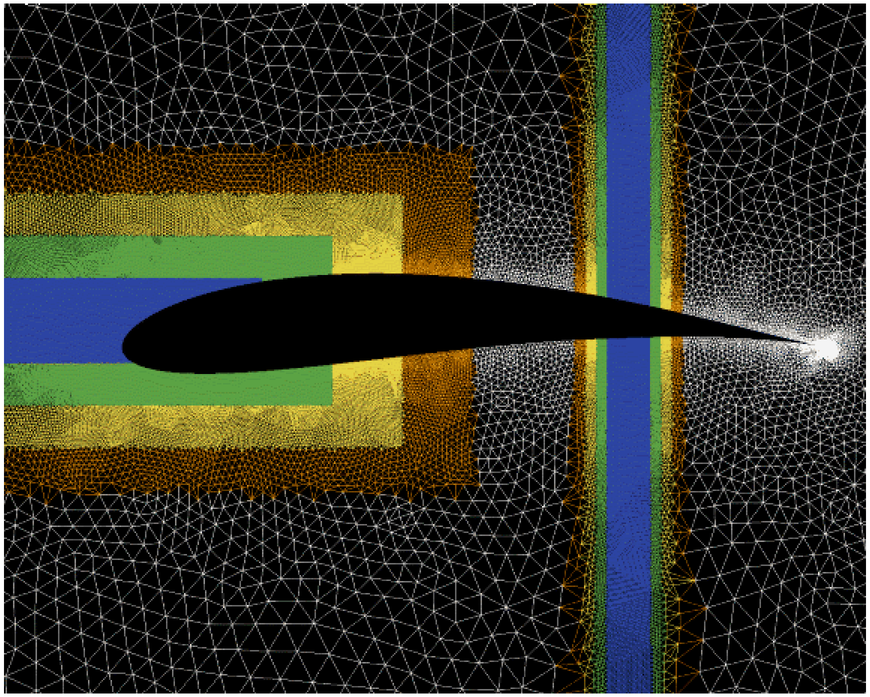
Unstructured Mesh Refinements

- Goal: Refine regions of a mesh to better capture fine-scale phenomena, or to handle stability and accuracy.
- Powerful tool for efficiently solving computational problems with evolving physical features (shocks, vortices, shear layers, crack propagation).

14,605 vertices
28,404 triangles

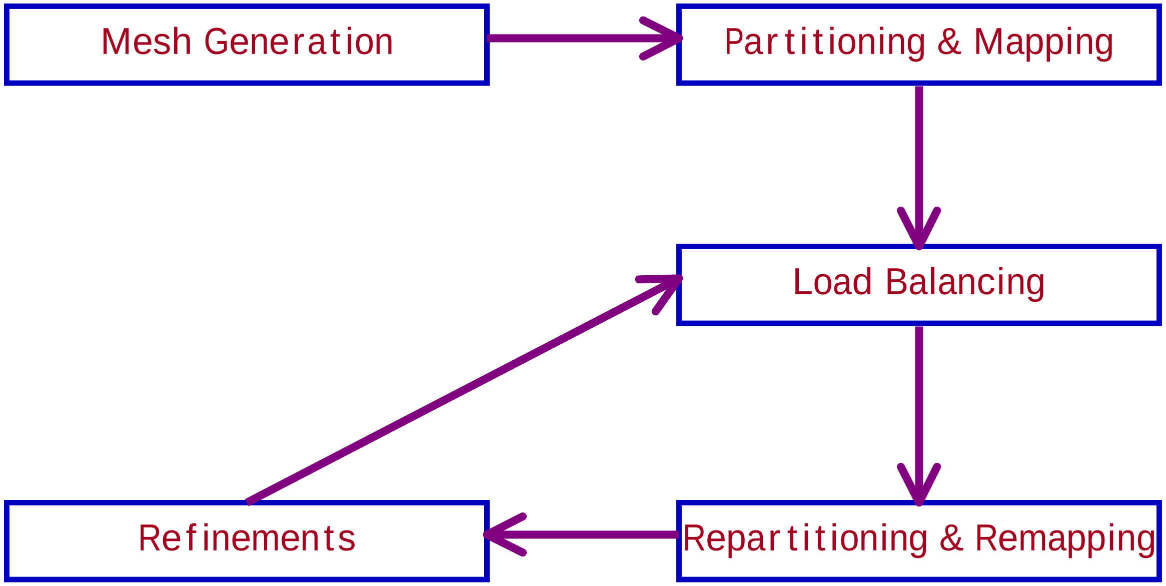


488,574 vertices
1,291,834 triangles



Unstructured Mesh Adaptations and Refinements

- Complicated logic.



- Many computer science issues.
  - Data structures.
  - Algorithmic choices.

Parallel Unstructured Mesh Adaptations and Refinements

- Difficult to parallelize efficiently.
  - Irregular data access patterns (pointer chasing).
  - Workload grows/shrinks at runtime (dynamic load balancing).
  - Workload redistribution (remapping).

- Developed PLUM and implemented on several architectures.
  - Cray T3E
  - SGI Origin 2000
  - Tera MTA

Program Paradigm	System	Best Time	P	Code Incr	Mem Incr	Scala-bility	Porta-bility
Serial	R10000	6.4	1				
MPI	T3E	3.0	160	100%	70%	Medium	High
MPI	O2K	5.4	64	100%	70%	Medium	High
Shared-mem	O2K	39.6	8	10%	5%	None	Medium
Multithreading	MTA	0.35	8	2%	7%	High*	Low

Numerical Linear Algebra

- Direct methods for solving sparse systems of linear equations.
  - Issues:
    - fill
    - data structures
    - algorithms
    - performance

- Have developed many state-of-the art solvers, which have been incorporated in various large-scale scientific and engineering applications.
- BlkFCT - solver for symmetric positive definite matrices.
  - optimization applications
  - structure analysis and structure dynamics calculations
  - computational fluid dynamics calculations
  - statistical analysis
- SuperLU - solver for general nonsymmetric matrices.
  - computational quantum chemistry
  - circuit simulations
  - materials science

Numerical Linear Algebra

- Iterative methods for solving systems of linear equations.
  - Issues:
    - convergence
    - preconditioning techniques
    - efficiency/performance
- Eigenvalue computations.
  - Issues:
    - convergence/accuracy
    - sparsity concerns
    - efficiency/performance
- Staff members @ NERSC are involved in the development of parallel preconditioning techniques.
  - structural analysis
  - materials science
  - image analysis
- Expertise in Lanczos/Krylov-type algorithms for eigenvalue computations.

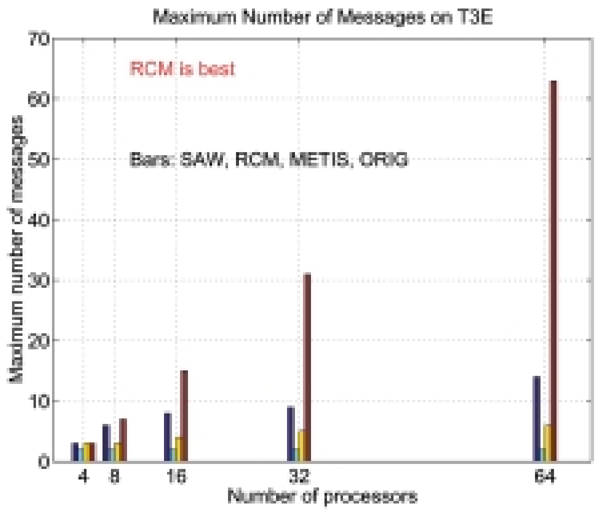
Recent Results in Sparse Matrix-Vector Multiplications

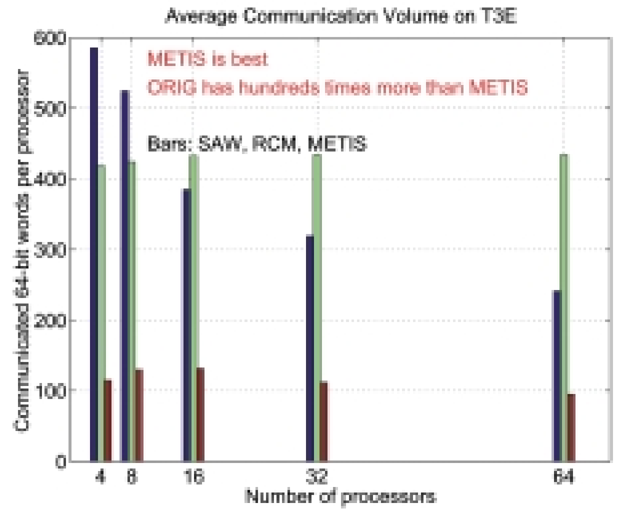
- Matrix-vector multiplication is a crucial kernel in iterative methods and generalized eigenvalue algorithms.

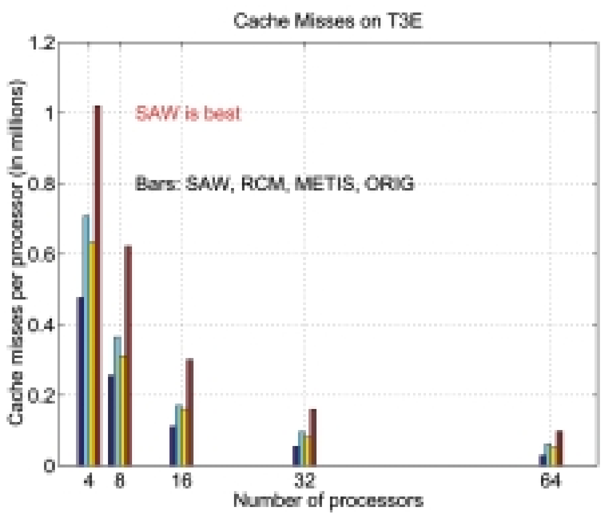
- “Ordering” of matrices (particularly for those arising from PDE-applications):
  - Determines the sparsity pattern of matrices.
  - Affects data access pattern in sparse matrix-vector multiplications.
  - Ordering time is crucial when mesh refinements are needed.

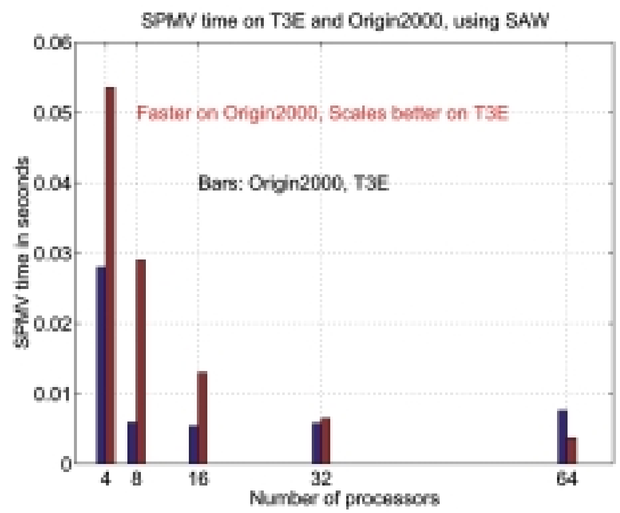
- Preliminary study using 4 ordering options:
  - Original order (ORIG).
  - Self-Avoiding Walk (SAW).
    - mesh-based linearization with excellent locality, especially attractive for mesh refinement
  - Reverse Cuthill-McKee (RCM).
    - reduce profile or bandwidth
  - Graph partitioning (METIS).
    - reduce number of edge cuts (communication)
- Test problem: 661,054 vertices and 1,313,099 triangles; assembled matrix has 2,635,207 nonzeros.

Recent Results in Sparse Matrix-Vector Multiplications









On-going Work on Sparse Matrix-Vector Multiplications

- Hybrid algorithms
  - Partition using good partitioner (e.g., METIS), followed by local reordering (e.g., SAW).
- Handle adaptively refined meshes.
- Integrate algorithms in iterative solvers (e.g., AZTEC) and eigen-solvers (e.g., Lanczos/Jacobi-Davidson eigenvalue algorithms).
  - Particularly important in accelerator modeling effort.

Never-ending Issues/Challenges

- Changing architectures …
  - Programming paradigms.
  - Combining shared-memory (e.g., OpenMP) and distributed-memory (e.g., MPI).
    - methodologies, strategies

- Increasing memory hierarchy ...
  - Data partitioning/locality/access.

- Problem-dependency …
  - Every problem has something different.
  - Algorithms need be adapted and/or designed accordingly.

- NERSC is committed to provide the expertise and to engage in a long-term research collaborative effort with its users.